From: 2001 IEEE Int'l Symp. on Semiconductor Manufacturing (ISSM) Conference Proceedings, San Jose, California, Oct 8-10, 2001, p. 107ff

Reconciling High-Speed Scheduling with Dispatching in Wafer Fabs

Mark D. Johnston
PRI Automation/Interval Logic Corporation
420 Bernardo Ave.
Moutain View, CA 94043 USA
mjohnston@ilc.pria.com

Abstract – A novel approach to reconciling global optimizing scheduling with real-time dispatching is presented. High speed scheduling is accomplished by two simultaneous cooperating scheduling processes, one of which generates longer term schedules while the other is focused with higher resolution on the nearer term. Real-time dispatching is based on the most recently generated near term schedule, carefully reconciled with events that could affect its current validity on the shop floor. Results are presented which show a fivefold performance speedup in schedule generation time, as well as significant schedule quality improvements over pure dispatching approaches.

1. INTRODUCTION

Improving the productivity of today's wafer fabs requires the convergence of many factors. Key among these is the automation software that monitors and controls overall fab operations, guiding the system as a whole to achieve its intended goals. For this to be effective, the most current information available must be combined with plans and schedules generated with the "big picture" in mind. This is extremely challenging, since the requirement for near real-time responsiveness is frequently in conflict with the desire to analyze and optimize facility operations. This paper shows how to reconcile the real-time requirements of dispatching with the global optimization available through scheduling.

In the following section (§2) we first define some key terms and then discuss some of the issues associated with scheduling and dispatching in a wafer fab environment. This is followed in §3 by a description of the software components that comprise the "dual span" architecture: two cooperating schedulers with real-time event reconciliation for dispatching. Results are presented in §4 along two dimensions of interest: runtime performance when generating new optimized schedules, and schedule quality by comparison with pure dispatching heuristics. We summarize our main conclusions in §5.

2. SCHEDULING AND DISPATCHING

The terms "scheduling" and "dispatching" are often interpreted differently by different people. Here we use the terms consistent with [1]: "scheduling" refers to the process of generating a set of future task assignments to times and resources, over some extended time interval, in order to meet various objectives. "Dispatching" means the process of deciding exactly which task(s) to execute when such a decision is called for, generally in a real-time sense. By this definition dispatching is clearly part of every manufacturing operation. However, the basis for the dispatching decision process varies widely. It is frequently derived from a set of heuristic rules which are expected to provide good guidance under typical circumstances. There have been numerous studies of potential heuristics relevant to semiconductor manufacturing (a recent example may be found in [2]).

The principal rationale for scheduling is to improve dispatching decisions and thereby ensure that the overall manufacturing process is better meeting global goals. These goals can differ from one facility to another but typically include, with varying degrees of importance, such items as: maximize throughput, on-time delivery performance, and utilization of key (bottleneck) equipment, while minimizing cycle time and its variance, and running expedited "hot" lots as fast as possible. See [3] for a more extensive discussion of many of these factors and their interrelationships.

Global scheduling which optimizes for these goals provides a number of well-known advantages over heuristic dispatching [4]:

- looking further into the future enables decisions that anticipate future events rather than only react to their occurence
- the consequences of decisions early in the scheduling interval can be better evaluated and modified in light of their downstream effects
- goal-driven scheduling is more flexible in the face of changing manufacturing conditions — there is no need for large rule sets to cover all contingencies
- interactions and tradeoffs among potentially competing factors can be naturally incorporated into the scheduling process

However, these potential advantages have previously been viewed as counterbalanced by some drawbacks:

- the time to generate an optimized schedule can be sufficiently long that the cumulative effect of changes in the fab render it no longer valid
- the unpredictable environment of the fab makes questionable how much benefit accrues from the effort invested in optimized scheduling

The longer the scheduler lookahead time, the greater the run time taken, and the more out-of-date the schedule will be when complete. It is this dilemma that we address as follows:

- split the schedule generation process into long- and short-term cooperating processes ("dual span") to dramatically speed up the time to generate new schedules
- reconcile in real-time the latest changes in the fab with the most recently generated schedule as the basis for dispatching

The next section describes the architecture and methodology we have developed.

3. DUAL SPAN SCHEDULING

The key architectural elements of the system are illustrated in Fig. 1. The Facility Model (FM) is a dynamically updated datastore that maintains the current state of the facility. It is kept up-to-date by a real-time event stream, generally from the fab's Manufacturing Execution System (MES). The FM also records data from the planning system, such as lot required completion dates and stage WIP and move rate targets.

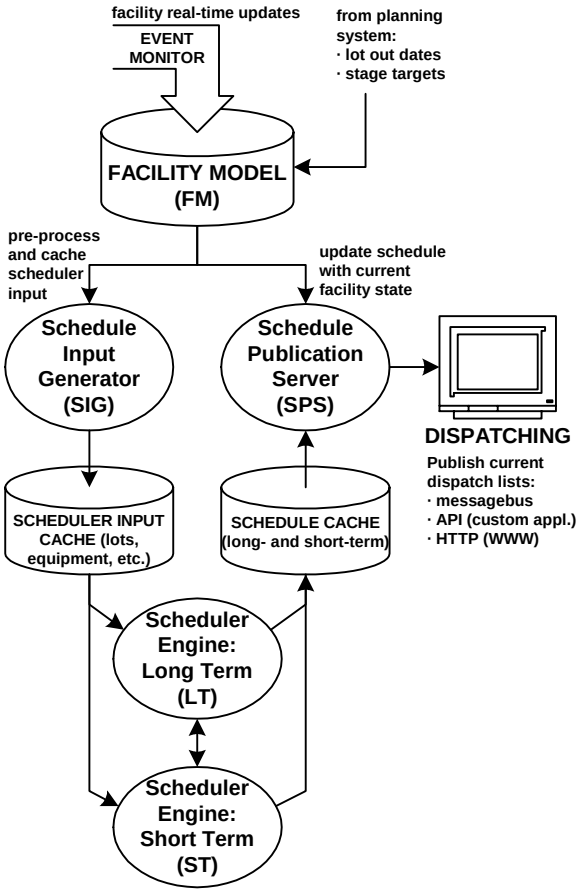

Fig. 1. Architectural overview of dual span scheduling.

The content of the FM is distilled and cached by the Schedule Input Generator (SIG), to minimize the lead time for schedule generation. Scheduling is performed by the two Repair Based Scheduling (RBS)[4,5] processes illustrated at the bottom of the diagram, which operate in a "dual span" mode [6]. Dual span scheduling is a unique approach to boost scheduling performance and accuracy by running two instances of the scheduler engine simultaneously. One of these (LT) is working over a longer time horizon with coarser time resolution and some filtering of tasks to schedule. The other (ST) is focused on nearer term scheduling, considering all of the required tasks and their resources, constraints, and preferences. The two scheduling processes communicate with each other: ST loads and uses the most recent long-term schedule as a constraint, while LT loads the most recent short-term schedule to use for initial condition continuity.

Typically the LT scheduler works on a somewhat abstracted version of the long term scheduling problem, possibly at coarser time resolution. Tasks which do not affect the global schedule significantly may be filtered out, leaving for full consideration:

- long preventive maintenance and batching tasks
- lithography tasks for which reticle management is important
- relative time critical tasks (e.g. "queue time constrained" tasks with a maximum allowed separation)
- tasks early in the schedule
- tasks at stage boundaries needed to assess line balance

Both LT and ST schedulers run continuously, getting updated facility information from the continually refreshed scheduler input cache. Schedules are saved to a schedule cache for fast access by other elements of the system. The cycle time for generating new short-term schedules is a few minutes, as discussed in the next section in more detail.

Repair-based scheduling (RBS) is a high performance scheduling technology that takes into account global optimization factors over a longer timescale, providing the powerful advantages over pure dispatching approaches described in §2. However, each generated schedule will soon become out-of-date in a typical fab environment, due to the occurrence of many unpredictable events. This is handled by the Schedule Publication Server (SPS) process, which continuously reconciles the latest generated schedule with the current state of the facility from the FM.

The schedule reconciliation process compares the scheduled task time and resource assignments with changes that have occurred since the start of schedule generation. For example, a lot may be scheduled but is then placed on hold: the SPS will ensure that it is removed from any dispatch lists on which it might have appeared. Other events that can immediately affect the schedule validity are handled similarly, such as equipment state changes, lot rework, track-in

on other than the scheduled equipment unit, etc. The time to process and reconcile the schedule after an event is received is at most a few seconds. As entirely new short-term schedules are generated by the ST process, SPS loads and reconciles them in the background, then switches over seamlessly to dispatching from the latest schedule. Only the reconciled schedule is published out to the fab as the dispatch list for operational use. This ensures that the dispatching data is consistent with whatever changes may have occurred, even very recently.

The architecture of Fig. 1 is implemented as a distributed system. Each of the scheduler engine processes runs on its own workstation to minimize contention for CPU, memory, and I/O resources. The remaining processes are allocated to a third workstation, with a fourth serving as hot standby for high availability operation support.

The combination of rapid optimizing schedule generation plus real-time reconciliation for dispatching addresses the key issues discussed in §2.

4. RESULTS

The architecture described in §3 has been implemented in the Leverage for Scheduling® software system. This section describes some of the results obtained from running the system.

Runtime Performance

Minimizing the schedule generation cycle time is key to ensuring that schedules, when published, are as up to date as possible. The dual span approach parallelizes the scheduling process, thereby reducing the latency of each newly generated schedule. The following table compares single and dual span run time performance on the same large dataset (based on actual fab data: 1850 lots with 44,600 tasks to schedule over a 24h period). All runs were made on the same current technology Windows 2000 workstation (Pentium IV 1.7Ghz processor).

Mode	Schedule duration/ time resolution	Time to schedule
Single	24h / 10m	15.2m
Dual Span	long term: 36h / 10m	6.2m
	short term: 6h / 5m	2.9m

In dual span mode the short term detailed schedule generation time, which is the relevant time for dispatch list generation, is about five times faster than single mode.

Schedule Quality Analyses

To investigate the important question of schedule quality in the face of frequent unpredictable disruptive events in a fab environment, a number of simulation studies have been conducted. These studies have been based on a smaller fab model, derived from the SEMATECH 300mm process flows, with the following characteristics:

- a representative 300mm processing flow of 244 steps utilizing 35 different equipment types, divided into 95 stages for wafer move accounting. The nominal flow cycle time was 20.6 days.
- a modeled WIP level of 200 lots of 24 wafers each, of two different products, with an approximate steady state starts rate of 10 lots/day. The initial WIP distribution was approximately uniform along the flow.
- 4% hot lots among both WIP and new starts
- due dates for all lots based on nominal flow cycle time
- a close to capacity equipment complement, i.e. 8 of the 35 equipment types had expected utilization >90%
- unscheduled machine downtime consistent with a 5% to 40% derating value depending on equipment type, and an 8 hour MTBF — this provides a very high frequency of disruptive equipment events
- 14 day simulation interval with 10 minute resolution

For comparison of repair-based scheduling **(RBS)** with a pure dispatching approach, two simple but widely used and robust dispatching heuristics were run through the identical simulation: critical ratio **(CR)** and first in/first out **(FIFO)**. Runs were evaluated using the metrics discussed in §1. The results are summarized in the following.

Move rate. The average stage move rate for each method is given in the following table. The rate for each was very close to constant over the entire 14d duration of the simulation, which lends confidence that initial conditions were not a significant perturbing factor.

Move Rate	RBS	CR	FIFO
Average daily stage move rate (wafer stage moves /day)	18,180	15,800	16,270

RBS achieves a move rate 15% greater than CR, and 12% greater than FIFO.

Cycle time. The median and standard deviation cycle time per step is given in the following table, for all lots with >10 completed steps during the simulation timespan.

Cycle Time	RBS	CR	FIFO
Median cycle time per scheduled step (hours)	2.29	2.58	2.72
Standard deviation in cycle time (hours)	1.40	1.72	1.52

The reduction achieved by RBS over the dispatching methods is 11-16% in cycle time and 8-19% in standard deviation of cycle time.

On time delivery. The cumulative distribution of lot completion time with respect to due time is shown in Fig. 2. The horizontal axis is scaled so that zero on the chart is the median of the RBS distribution (+5 hours later than the actual time due). Of the dispatching rules, CR does much better than FIFO, as expected. However, RBS does substantially better than CR: when 50% of the RBS lots are out, only 25% of those run with CR have completed.

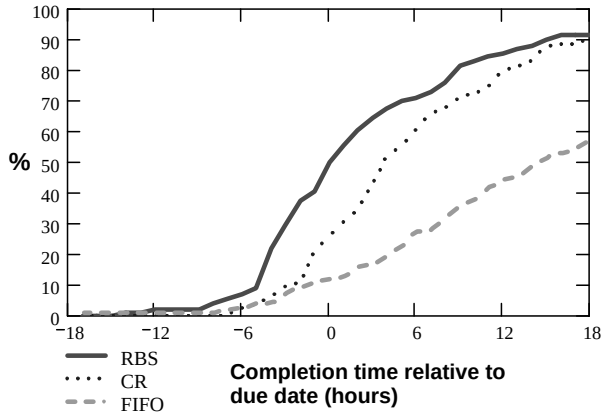

Fig. 2. On-time delivery performance: the distribution of completion time relative to due date for the lots in the simulation study

Batching performance. The following table shows total wafer moves on the major batch tools, and the mean batch size in wafers averaged over all batched steps.

Batching Performance	RBS	CR	FIFO
Total wafer moves on batch tools	42,384	38,352	39,360
Mean wafers/batch	95.2	87.2	87.7

RBS is able to both build bigger batches (by 9%) and schedule more batch moves (by 8-11%).

Setup avoidance. In this simulation model the implanters required a setup time of 20-30 minutes for species change. The total scheduled setup time for each method is given in the following table.

Setup Time	RBS	CR	FIFO
Total scheduled implanter setup time (hours)	76.8	159	160

RBS reduced the total setup time by about a factor of two.

"Hot Lot" performance. Hot lots (4% of the total in this model) act as a significant perturbing influence, but their expedited movement is an important operations objective. The following table summarizes the results in terms of stage moves/hour for all hot lots running in the simulation.

Hot Lots	RBS	CR	FIFO
Total stage moves per hour	54.6	49.9	49.5

RBS moves these critical lots at about a 10% faster rate than the dispatching heuristics (which always sort hot lots as top dispatching priority).

5. CONCLUSIONS

In this paper we have demonstrated a practical mechanism for reconciling global optimizing scheduling and real-time dispatching:

- very high speed schedule generation times can be achieved by distributing the problem over two cooperating parallel scheduling processes ("dual span")
- accurate and timely dispatch lists can be generated by reconciling in real-time the optimized schedule with recent fab events

Simulation results show that, even in the face of a high rate of unpredictable events, the schedule quality achievable with repair based optimization is considerably better than that observed with some commonly used pure dispatching heuristics. This applies to metrics on many dimensions at once, even to those often thought of as competing.

REFERENCES

[1] *Scheduler/Dispatcher User Requirements*, International SEMATECH Tech. Transfer MET202-Draft, Feb 2000

[2] J-J. Chen et al., "Real-Time Dispatching Reduces Cycle Time", *Semiconductor International*, March 2000

[3] H. Watts, "Improving Fab Performance", *Future Fab International* Vol. 9, pp. 98ff, 2000

[4] H. Watts and M. D. Johnston, "Fab Production Schedules", *Solid State Technology*, July 2001

[5] M. D. Johnston and S. Minton, "Analyzing a Heuristic Strategy for Constraint Satisfaction and Scheduling", in *Intelligent Scheduling*, ed. Zweben & Fox, 1994

[6] M.D.Johnston,"Scheduling Tools for Astronomical Observations", in *Proc. Conf. New Observing Modes for the Next Century*, ed. Boroson, Davies & Robson, 1996

AUTHOR BIOGRAPHY

Dr. Mark D. Johnston is Chief Technology Officer of Interval Logic Corporation, a PRI Automation company. He received his BA and PhD degrees from Princeton University and MIT, respectively, and has worked and published extensively in the areas of advanced planning and scheduling.